(12) UK Patent Application (19) GB (11) 2 316 503 (13) A

(43) Date of A Publication 25.02.1998

(21) Application No 9617596.3

(22) Date of Filing 22.08.1996

(71) Applicant(s)
ICL Personal Systems Oy

(Incorporated in Finland)

PO Box 458, SF-00101 Helsinki, Finland

(72) Inventor(s)
Tapani Lindgren

(74) Agent and/or Address for Service
S M Dupuy
International Computers Limited, Cavendish Road,
STEVENAGE, Hertfordshire, SG1 2DY,
United Kingdom

(51) INT CL^6
G06F 1/00

(52) UK CL (Edition P)
G4A AAP

(56) Documents Cited
GB 2236604 A EP 0332304 A2 WO 93/11480 A1
US 5375206 A US 4924378 A

(58) Field of Search
UK CL (Edition O) G4A AAP
INT CL^6 G06F

(54) Software licence management

(57) A software licence management method and system is for a computer system including at least one server (1,5) and particularly for a plurality of computers connected via a network. Before a service (2) can offer functionality to a user it has to check that the user has a licence for that service. A licensing subsystem (3) has associated with it a ticket database (4) that hold tickets corresponding to existing licences. Tickets, if available, are issued to a service on request, thereby verifying the existence of a licence. The receipt of a ticket allows a service to offer functionality.

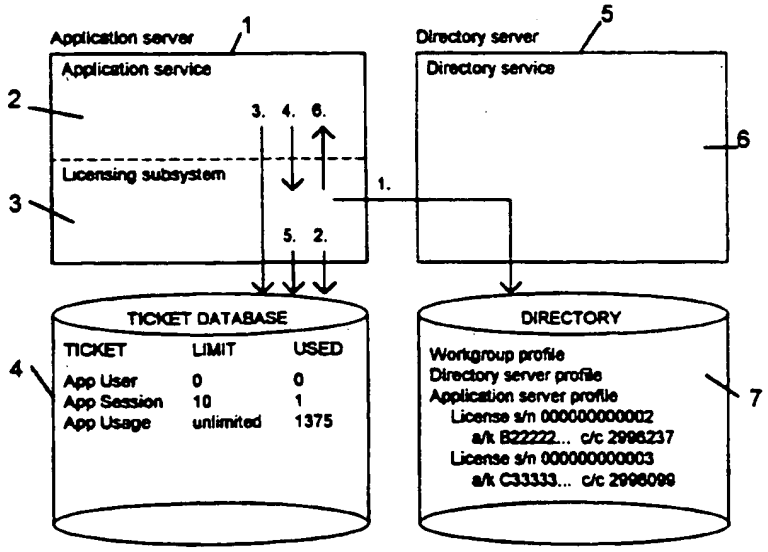

FIG 1

GB 2 316 503 A

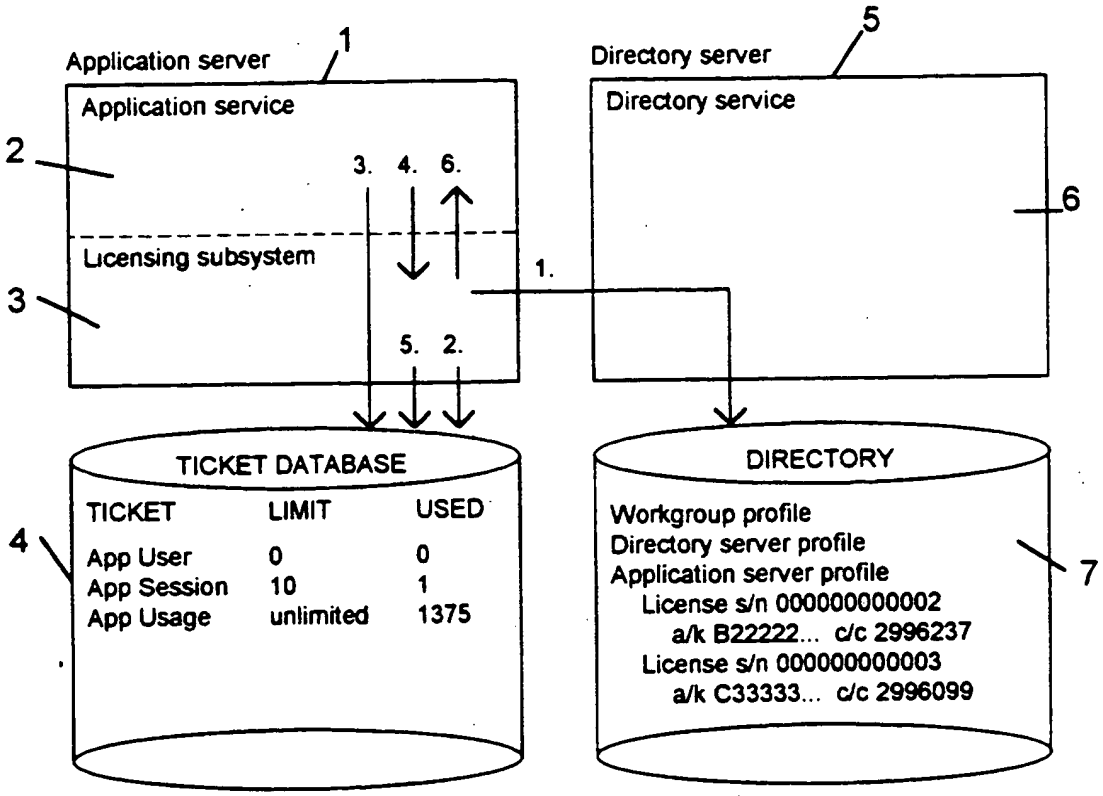
Application server
1
Application service
2
3.
4.
6.
Licensing subsystem
3
1.
5.
2.
TICKET DATABASE
TICKET
LIMIT
USED
4
App User
0
0
App Session
10
1
App Usage
unlimited
1375
Directory server
5
Directory service
6
DIRECTORY
Workgroup profile
Directory server profile
Application server profile
License s/n 000000000002
a/k B22222... c/c 2996237
License s/n 000000000003
a/k C33333... c/c 2996099
7

FIG 1

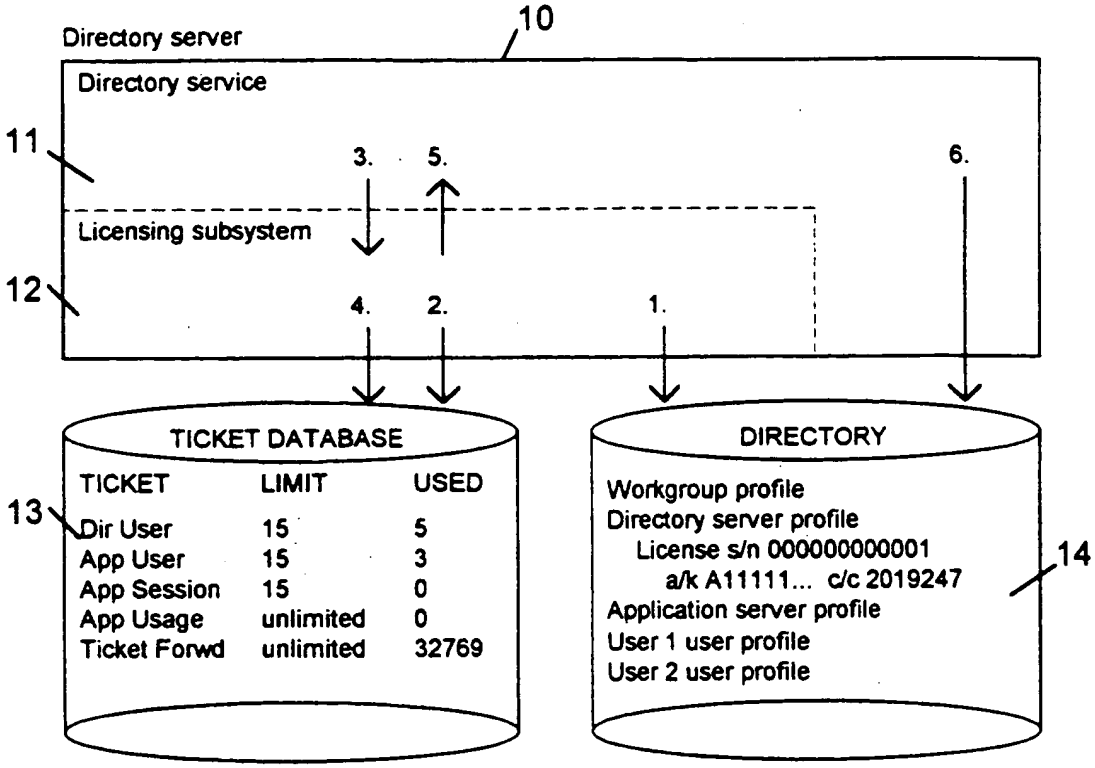
Directory server
10
Directory service
11
3.
5.
6.
Licensing subsystem
12
4.
2.
1.
TICKET DATABASE
TICKET
LIMIT
USED
13
Dir User
15
5
App User
15
3
App Session
15
0
App Usage
unlimited
0
Ticket Forwd
unlimited
32769
DIRECTORY
Workgroup profile
Directory server profile
License s/n 000000000001
a/k A11111... c/c 2019247
14
Application server profile
User 1 user profile
User 2 user profile

FIG 2

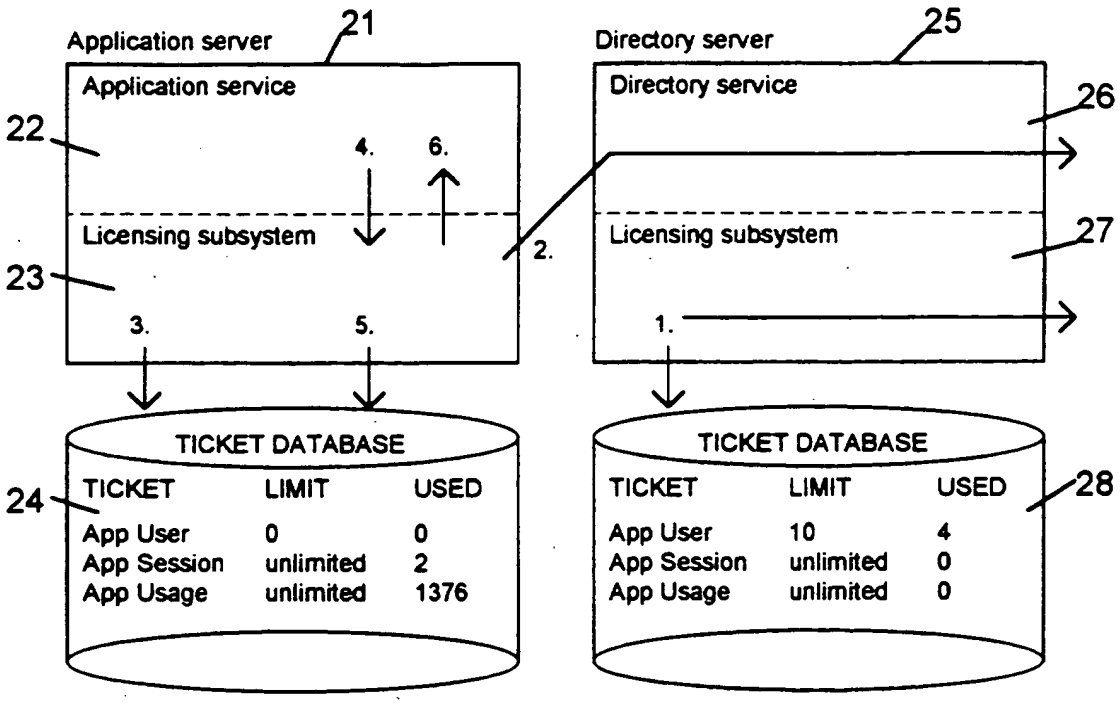
Application server
21
Application service
22
4.
6.
Licensing subsystem
23
2.
3.
5.
Directory server
25
Directory service
26
Licensing subsystem
27
1.
TICKET DATABASE
TICKET
LIMIT
USED
App User
0
0
App Session
unlimited
2
App Usage
unlimited
1376
24
TICKET DATABASE
TICKET
LIMIT
USED
App User
10
4
App Session
unlimited
0
App Usage
unlimited
0
28

FIG 3

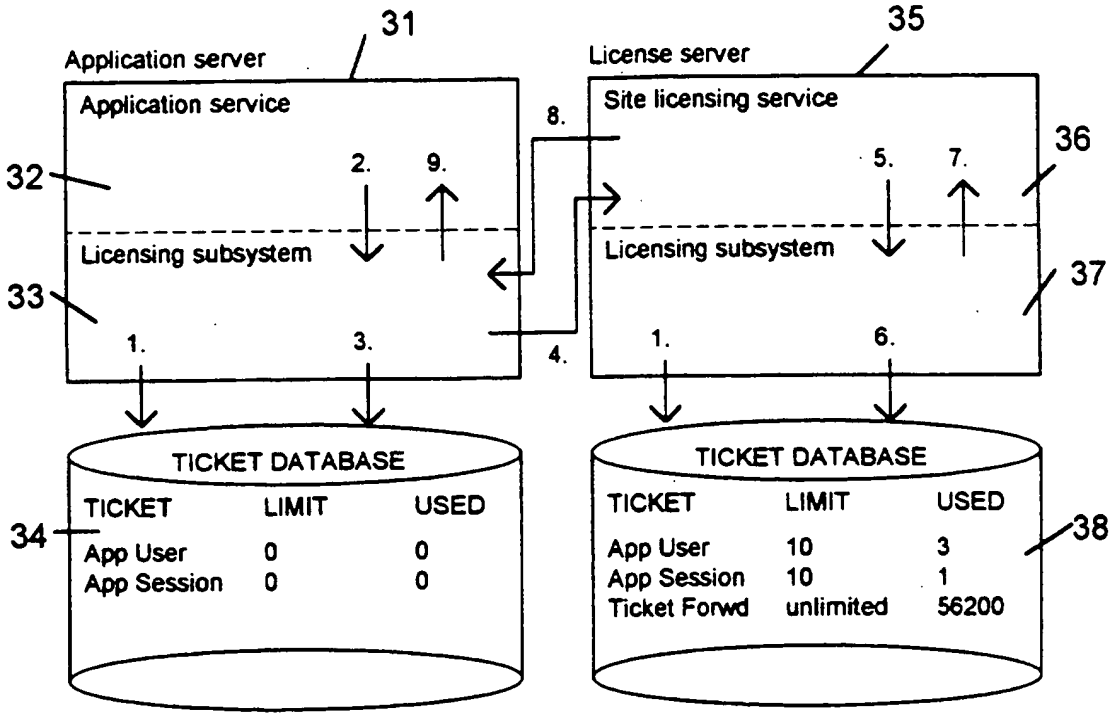

FIG 4

2316503

SOFTWARE LICENCE MANAGEMENT

This invention relates to software licence management and in particular to licence management for software running on a plurality of computers connected via a network.

Conventionally, licences have been provided by software vendors as separate licences for individual workstations or as a single licence for a number of workstations. Various schemes have been proposed in order to try and make unlicensed software unusable, in particular pirated (illegal) copies of software. Other schemes have been proposed such as in order to achieve low initial software costs but licensing royalties consistent with the extent of use, in order not to deter low-usage users from purchasing particular forms of software, and thus to reduce piracy, whilst still enabling a vendor to collect higher dues from high-usage users.

The present invention is, particularly, concerned with a distributed system consisting of various server and client programs running on various computers which are connected via a local or wide area network, and an object is to provide server software licensing which ensures that all software running in the network has been purchased legally.

According to one aspect of the present invention there is provided a software licence management method for use with a computer system including at least one server, the method being such that before a service can offer functionality to a user, the said service shall verify that the user has a licence for said service, and wherein the computer system further includes a licensing subsystem with which are associated service tickets corresponding to existing licences, the method including the steps of the said service requesting a respective service ticket from the licensing

subsystem prior to offering functionality to the user, and the licensing subsystem issuing a said service ticket to the said service, if one is available, thereby verifying the licence exists and allowing the said service to offer functionality.

According to another aspect of the present invention there is provided a computer system including at least one server and a software licence management system, the management system being such that before a service can offer functionality to a user, the service shall verify that the user has a licence for said service, the management system including a licencing subsystem with which are associated service tickets corresponding to existing licences, and the management system being such that a said service ticket is issued to a service, if one is available, upon request by the service, thereby verifying existence of a licence and allowing the said service to offer functionality.

Embodiments of the invention will now be described with reference to the accompanying drawings, in which:

Figure 1 Illustrates obtaining a session or usage ticket for an application,

Figure 2 Illustrates obtaining a user ticket for a directory server,

Figure 3 Illustrates independent licence sharing, and

Figure 4 Illustrates licence sharing with a site licencing service,

Various terms used in the following will first be defined. For the purposes of the description the software is considered to relate to a Groupware Office system which provides various facilities including mail, for example.

Definitions

"Server"	An instance of server software running on a server computer. Usually only one such instance runs on any one computer. Each "server" implements one or more collections of related functions called "function sets", examples of which are directory, mail, library etc. The "directory function set" includes functions to access a database that contains information about the Groupware Office system.
"Client"	Any piece of software that connects to the "server" using a "client-server protocol" to access the functions offered by the "server". A "client" may be a program run by a user on a workstation, or a part of any other program.
"Session"	An instance of client-server dialogue between one "client and one "server". Each "session" allows the "client" to use the functions of one or more "function sets".
"Directory Server"	A server that implements the directory function set.
"Mail Server"	A server that implements the mail function set. [A server may be a directory server and a mail server simultaneously.]
"User"	A person (actual or virtual) listed in the database of a directory server.

"User profile" The information pertaining to one user stored in a directory database entry, such as the name of the user, user authentication information, the list of servers and function sets the user is permitted to access, etc.

"Server profile" The name and network address of a server and the list of services offered by it, as stored in a directory database entry.

"Service profile" Information stored in a directory database entry about one service in one server. If the same type of service is offered by more than one server, each instance has its own profile.

"Site profile" Information stored in a directory database entry about one site.

"Site" A set of servers connected to a single directory server. Each server belongs to exactly one site, and each site has exactly one directory server. Other servers in the site are optional, usually unlimited in number, and sometimes called member servers or application servers.

"Enterprise" A set of sites that share their directory databases. The directory servers in each site replicate directory information to other directory servers in the enterprise. Each directory server contains both "local" and "external" information. One of the directory servers, the "enterprise directory server" controls the others, which are

"site directory servers".

"Subsystems" — Collections of programs and/or subroutines that perform a set of interrelated functions. Some subsystems implement a function set within a server program, while others run independently as stand-alone applications. Many subsystems are collections of common subroutines called by other subsystems. Client programs are also subsystems.

"Subsystem id" — A respective unique number identifying every type of subsystem. Some systems use two ids, a "real subsystem id" when dealing with licensing issues and an "alias subsystem id" when performing a task on behalf of a virtual entity, such as "generic gateway no 9".

Not every subsystem software needs to be purchased individually. Most collections of subroutines can be used freely by other subsystems.

"Services" — Those subsystems that need to be explicitly purchased.

Service types may include directory service, mail service, fax gateway, X.400 gateway, enterprise option, library service, power library option, etc. Each service is located in one server, either as a function set of the server program or a standalone application running in the same computer. Many services of the same type can exist in different servers.

The software licence management system of the present invention proceeds from the premise that before offering any usable functionality to the users, services shall verify from a "licensing subsystem" that a licence for the service exists. To achieve that, it is proposed that the licensing subsystem holds "service tickets" and a service requests a permission to offer its functions to the user by requesting a corresponding "service ticket" be provided from the licensing subsystem. Each service knows that kind of tickets are needed to fulfil the service's functionality. The licensing subsystem has to keep track of the available licences and of the service tickets it has issued. A service ticket may be considered as partially the equivalent of a password in that one must be provided before a service can operate.

A "licence" is a permission to use one or more services within certain limits. Typically these limits are "license duration", which specifies the maximum length of the period when the licence may be used (the "active" period) and the "licence size", which specifies the maximum number of users of the licence. The interpretation of "number of users" varies from service to service. It may, for example, mean the number of users in the local directory that are allowed to use the service, or the number of concurrent sessions that are connected to the service. Licence duration and/or size may also be unlimited.

When a customer purchases a Groupware, for example, software product which employs the software licence management method and system of the invention from a supplier, as well as the media containing the software itself and associated documentation, there is obtained a single licence to one or more services. Each said product has a unique serial number. The license is supplied in the form of a licence agreement document on which the licence information is printed. This licence information consists of the serial number of the product and an "activation key" for the licence. The licence

size and duration and the included services are encoded into the activation key.

The software license management method and system of the invention is such that the Groupware software may be copied and installed by the customer without any technical restrictions, but before any of the services can be used, a corresponding licence must be installed and activated. Licence installation consists of entering the license information (serial number and activation key) into the server profile of a server in the directory server's database, ie in the site directory, in the server profile of the server in question. Licence activation consists of setting the active period of the installed licence so that service tickets can be issued. Typically licence installation and licence activation are performed simultaneously by the server setup program. The license information is stored in the site directory, in the server profile of the server in question.

As will be appreciated, there also exists "evaluation licences" which allow a prospective customer to use a service for a short trial period before actually purchasing it. These licences typically have a very short duration and a relatively small size. The product serial numbers associated with such evaluation licences are not necessarily unique, since the licence information may be distributed on CD-ROMs or via public networks.

As mentioned above, each software product contains just one licence, although that licence may include a large number of services, for example, enough to build a complete Groupware Office site with all of the basic services. Alternatively, the licence may include just one service. Product with that kind of licence could be used to expand the capacity or functionality of an existing Groupware Office System.

The core of the process of designing a product is, therefore, determining what services will be included and the size and duration of the licence. This information is encoded into a number, the covert code, which may be a 7-digit number, for example. The building of the covert code is discussed in more detail hereinafter.

The amount of information that can be encoded into the covert code is limited by the size of the code. Therefore, there are some necessary restrictions on what kinds of licences are possible. The most obvious limitation is that the size and duration can only take certain discrete values. Also, the same size and duration will apply to all services covered by the licence. Another restriction is that only the most common groups of services can be combined freely into a multi-service licence. Other services will have to be licensed individually.

The covert code, which specifies the properties of the software licence, is thus a part of the product description in the logistics database. When the product is manufactured it has the unique serial number, referred to above, assigned to it. The actuation key for the license is calculated as a function of the serial number and the covert code using a secret algorithm. The serial number and activation key may be printed on a label, which is attached to the licence agreement document.

When creating a site, a customer must have a licence that includes a site creation ticket. This licence is installed for the directory server. The customer may also install additional licences for the directory server and for other servers. Each licence may apply to one or more services. Some licences are valid only in that server for which they are installed, whilst other licences may be shared with other servers at the same site (see later). Shared licences would usually be installed in the server profile of the directory

server, although optionally, and with some restrictions, another server may be designated as the licence server. The serial number of the first licence installed in a directory server's profile can be used to identify the site uniquely. Thus the directory server is computer number 1. Other servers in the site will use the same site id but differing computer numbers for identification.

When a service program is about to execute an action which requires that a customer possesses a licence for that service, the service program must first obtain a corresponding service ticket from the licensing subsystem. The actions concerned are ones which are potentially profitable for the customer and may, for example, include namely:

setting up a new Groupware site;

creating a new user account;

setting up an instance of the mail service;

enabling mail usage for a user and creating a user mailbox;

starting a session between a mail UI client and the mail server;

sending a mail message;

relaying a mail message to an X.400 mail network.

Each kind of action requires a specific kind of service ticket. To obtain a ticket the service needs to specify the ticket type and the number of tickets. The service tickets are only identified by ticket type. There is a licensing subsystem in each server and it counts the number of

different tickets in all available licences and keeps track of how many licences of each type are being used in the server.

The steps involved in obtaining various licences will now be described in greater detail. With respect to Figure 1 there will, firstly, be described the case of an application service running in a separate server from the directory server obtaining a session or usage ticket.

Figure 1 illustrates schematically an application server 1, providing an application service 2 and having a licensing subsystem 3, with an associated ticket database 4, a directory server 5 providing a directory service 6 and having an associated directory 7. The ticket database 4 has stored therein details of ticket types, the limit, if any, of the number of such tickets which are available and the number of used tickets for each type. The ticket types as illustrated are "App User" (Application User), "App Session", "App Usage". The directory 7 has stored therein, the "Workgroup profile", the "Directory server profile", the Application server profile. In the example illustrated, the application server has two associated licenses whose serial numbers (s/n) are 000000000002 and 000000000003, respectively, whose activation keys (a/n) are of the form B22222... and C33333..., respectively, and whose covert codes (c/c) are 2996237 and 2996099, for example, respectively.

When the licensing subsystem 3 on the application server starts, it fetches the application server's server profile from the directory service 6, 7 using the directory API (Application Programming Interface) (Step 1 in Figure 1). The licensing subsystem 3 analyses the licences and updates the limits of each ticket type in the local ticket database 4. The numbers of used tickets are not modified at this time, the old accumulated values being maintained (step 2).

When the application service 2 starts, and before any user logs in, it tells the licensing subsystem 3 to set the number of used session tickets to zero. This frees any session tickets that may have been left unreturned at the end of a session because of a system crash etc. (Step 3). The application service 2 then requests a service ticket (session or usage) from the licensing subsystem 3, since without a ticket it cannot proceed. (Step 4). The licensing subsystem checks the ticket availability in the local ticket database 4 and updates the used ticket count (step 5), to take into account the requested ticket, before issuing the ticket to the application service (step 6), which then proceeds since it has determined that there exists the appropriate licence.

In the embodiment of Figure 2, the procedure whereby a directory service obtains a ticket for adding a user to a directory is illustrated.

A directory server 10 provides a directory service 11 and includes a licensing subsystem 12 with an associated ticket database 13, the directory service 11 having an associated directory 14. The ticket database 13 has stored therein details of ticket types, the limit, if any, of the number of such tickets which are available, and the number of used tickets for each type. The ticket types are illustrated as "Dir User" (Directory User), "App User", "App Session", "App Usage" and "Ticket Forwd" (Ticket Forwarding). The directory 14 has stored therein the "Workgroup profile", the "Directory server profile", the "Application Server profile" and the user profile of two users, User 1 and User 2. The Directory server has a license serial number (s/n) 000000000001, with an actuation key (a/) of the form A11111..., and a corresponding covert code (c/c) 2019247, for example.

When the licensing subsystem 12 on the directory server 10 starts, it fetches the directory server's server profile directly from the directory 14 (step 1). The licensing

subsystem 12 analyses the licenses and updates the limits of each ticket type in the ticket database 13. The numbers of used tickets are not modified at this time, the old accumulated values being maintained (step 2).

When it is desired to add a new user to the directory, the directory service 11 requests a user ticket from the licensing subsystem 12 (step 3). The licensing subsystem 12 checks ticket availability in the local ticket database 13 and updates the used ticket count (step 4) to take into account the requested ticket. The licensing subsystem 12 issues the requested user ticket to the directory service 11 (step 5). The directory service then adds the new user to the directory 14, that is it adds its user profile.

To ensure consistency, the directory service 11 may periodically count the number of users in the directory 14 and tell the licensing subsystem 12 to set the used ticket count accordingly.

When a licence is installed, the start time of its active period will be fixed. By default this is the same as the installation time, but any time in the past or in the future may be specified. If the licence has a limited period, the end time will also be set. The licence will be active whenever the current time is after the start time and before the end time.

A customer may wish to deactivate a licence so that it cannot be used. Thus can be done at any time by altering the end time of the licence with the server setup program. The end time may be altered freely, as long as the active period does not exceed the licence duration.

Once installed, limited-duration licences are fixed, ie they cannot be removed, except by remaining the entire site directory, or moved to another server, and their start time

may not be altered. These restrictions, however, do not apply to unlimited-duration licences. They may be removed, reinstalled, moved or altered freely. The only restriction that remains is that a licence may only be installed for one server at a time.

A further restriction applies to the licence that has been used to create a site. This licence cannot be removed or deactivated, except by removing the entire site.

The licensing method described with reference to Figures 1 and 2 applies only to local licences, ie the tickets included in a license can only be issued in one server, the server whose server profile contains the licence. Often there is a need to share a single licence between two or more servers, so that tickets can be issued in all of them. Most commonly, the user tickets for an application are needed in the directory server, and session and usage tickets in the application servers.

If a licence includes an unlimited number of a certain kind of service ticket, sharing the licence is not very complicated. Any server can read the licences in any other server's profile. If the licensing subsystem in a server can verify that another server's licence contains an unlimited supply of freely shareable tickets, it will deduce that these tickets may be issued without limit in any server, independently of other servers. This is independent license sharing.

Not all licences are necessarily shareable, even if they contain an unlimited number of tickets. Whether each licence is shareable or not is a licence-specific property, which is coded in the covert code together with other licence properties.

The first implementation of the licensing subsystem capable

of independent licence sharing will not scan every server profile for available licences. It will only scan its own server profile and the directory server's profile. Therefore, all licenses that are meant to be shared, should be installed for the directory server.

If the tickets to be shared are limited in number, the situation is more complicated. For each "pool" of shareable tickets, there must be a single process that is responsible for keeping track of their usage. It has to co-ordinate the activities of the licensing subsystems in various servers and make sure that no ticket is issued more than once. To achieve this a site licensing service can be implemented. This is an extension to the licensing subsystem that allows the licensing subsystems of various servers to communicate using a client-server protocol. The site licensing service, together with the licensing subsystem in the same server, control the usage of tickets installed for that server. Another server's licensing subsystem may connect to the site licensing service and ask the latter to obtain a service ticket on its behalf.

Licenses that are shareable by independent sharing would also be shareable by the site licensing service, with the addition that also limited-number tickets could be shared. Some types of licences will still be unshareable, since shareability is a licence-specific property. The licensing service could itself require a licence. A site licensing service could be expanded to support also client licensing and enterprise-wide licence sharing.

An example of independent licence sharing will now be described with reference to Figure 3 in which an application server 21 provides an application service 22 and includes a licensing subsystem 23 with an associated ticket database 24. A directory server 25 provides a directory service 26 and includes a licensing subsystem 27, with a associated ticket

database 28, and a directory (not shown but containing information of the type illustrated in Figures 1 and 2). The ticket databases 24 and 28 have details of ticket type, limit and usage as indicated.

The licensing system 27 on the directory server 25 fetches the server profile from the directory (not shown), analyses the licences therein, and updates the ticket limits (step 1). The licensing system 23 on the application server 21 fetches the application server's server profile from the directory (not shown) using the directory API. It also fetches the directory server's server profile (step 2).

The Application server's licensing subsystem 23 analyses the licences in the server's own profile. In this case there are none, since the example is concerned with licence sharing. The licensing subsystem 23 then analyses the directory server's licences. Because there are unlimited session and usage tickets in a shareable licence, the local limit is also set to unlimited. The user ticket limit is set to 0, because they are limited (10 according to ticket database 28) and limited tickets cannot be shared with this method (step 3).

The application service 22 then requests an application session ticket from its licensing subsystem 23 (step 4). The ticket is granted because there are an unlimited supply of them. The used ticket count is updated in the local ticket database 24 (step 5), although it is only needed for statistics as the number is unlimited. The session ticket is then issued to the application service 22, which then proceeds since it has determined that there exists an appropriate licence.

License sharing in the case of a site licensing service will now be described with reference to Figure 4, in which an application server 31 provides an application service 32 and includes a licensing subsystem 33 with an associated ticket

database 34. A license server 35 provides a site licensing service 36 and includes a licensing subsystem 37 with an associated ticket database 38.

The licensing subsystems 33 and 27 of the servers 31 and 35, fetch their corresponding server profiles from a directory (not shown), analyse installed licences and store the ticket limits in the local databases 34 and 38 (step 1). The application server 31 need not have any licences.

The application service 32 requests a service ticket, for example an application session ticket, from the local licensing subsystem 33 (step 2). The local licensing subsystem 33 in the application server 31 will first attempt to issue the ticket locally, but this will fail as there are no licences installed for the application server 31, as indicated by the lack of available tickets in the ticket database 34 (step 3). The licensing subsystem 33 in the application server 31 will then connect to the site licensing service 36 using the client-server protocol and request the ticket remotely (step 4). The site licensing service 36 requests the ticket from the local licensing subsystem 37 and it also request a ticket-forwarding ticket (step 5). The licensing subsystem 37 of the license server 35 checks ticket availability and updates the used ticket counts in the ticket database 38 (step 6). The tickets are issued to the site licensing service 36 (step 7) which forwards the application ticket to the client ie licensing subsystem 33 (step 8), which as a result issues the application ticket to the application service 32, allowing that to proceed (step 9).

Whenever a licensing subsystem issues a service ticket, or a ticket is returned such as because it is an unused ticket (any number can be requested) or because it is a session ticket, which are required to be returned at the end of a session, the transaction can, optionally, be logged to a log file which is separate from other log files in the system.

The information in this separate log file may be used to implement a pay-by-usage licensing scheme (delayed billing). Logging can be enabled or disabled by an administrator. Each server has its own log file and all kinds of tickets issued in the server will be logged the same way. Logging parameters for each kind of ticket could be specified for certain types of licences, although such a licence could not be shared by the independent sharing method.

The proposed licensing method allows for introducing new services while retaining compatibility with old licences. The licensing subsystems will initially support some types of licences and service tickets that are not yet connected to any particular service. New services can be assigned to these items without making any modifications to existing administration programs and the licensing subsystem. The method could be extended further by adding new license/ticket combinations to the licensing subsystem, although all existing combinations would need to be kept unchanged. This would involve updating the licensing subsystem in all servers where the new services would be used. Older subsystems would not accept the new kind of licenses not issue tickets for the new services. The licenses and tickets could be defined statically, as they are now, although there could be other possibilities.

As discussed above, the covert code specifies the licence duration, licence size and included services. An example of a covert code comprises a 7-digit decimal number, with the digits numbered from right to left, starting from zero eg in number 6543210, digit no 0 is "0", digit no 1 is "1" etc.

Licence duration may be encoded in the last digit ie digit 0, as follows:

Digit No 0	Licence Duration
"0"	10 days
"1"	1 month (31 days)
"2"	3 months (92 days)
"3"	6 months (184 days)
"4"	1 year (366 days)
"5"	2 years
"6"	3 years
"7"	Unlimited (small size)
"8"	Unlimited (medium size)
"9"	Unlimited (large size)

Licence size may be coded in the next-to-last digit, digit no 1. However, its interpretation may depend on the licence duration. Limited duration licences may be one of, for example, 30 different sizes; duration digits "7", "8" or "9" select small, medium or large licence sizes respectively.

Digit No 1	Licence size for each duration type			
	Limited	Unlimited Small	Unlimited Medium	Unlimited Large
"0"	1	1	60	400
"1"	2	2	80	500
"2"	5	5	100	600
"3"	10	10	125	800
"4"	15	15	150	1000
"5"	20	20	175	1200
"6"	30	25	200	1500
"7"	50	30	225	2000
"8"	100	40	250	3000
"9"	Unlimited	50	300	Unlimited

The services that are included in a licence may be encoded into four digits, digits no 2 to no 5, of the covert

code. These digits are called the service code. The licence may apply to one kind of service tickets only, to a group of related service tickets that are used by one service, or to a group of selected services. The service code can be chosen to represent a particular name of service, such as "basic directory service", "basic mail service", "basic calendar service", in any desired manner but it will indicate what types of tickets are included and how many licence service tickets are included for each type of service.

The digit no 6, the most significant digit, may be used to specify a particular product line. In the examples shown in the drawings the covert codes all commence with the number 2, indicating they relate to the same product line.

Any number of licences may be installed in the server profile of any server. The activation key is verified, and the covert code calculated from the serial number and the activation key at license installation time. The mapping of covert code to service ticket is, preferably, not stored in the directory, rather it is recalculated by a licensing subsystem every time it starts up. All tickets of the same type are indistinguishable. The licensing subsystems do not keep track of individual tickets issued.

Any number of identical tickets may be obtained at once by a service from the corresponding licensing subsystem, providing of course that they are available. Tickets can be returned if they are not used.

The licensing subsystem does not force services to obtain tickets rather it is the service's responsibility to offer services only to legal users and without obtaining a respective ticket, a service which requires a licence will not function.

Session tickets are associated with client-server sessions.

Unless a service wants to allow unlimited usage, it should obtain a session ticket whenever a session starts. Determining when each session starts and ends is the responsibility of the service. Session tickets may not be applicable to all services. It is important that session tickets are returned when the sessions end, otherwise they will be unusable, at least until the licencing subsystem is resynchronised. This is achieved at server start up, when there are no sessions in existence, by setting the used session ticket count to zero.

When a user is given the right to use a service, the associated user ticket should be obtained first. Because in a currently preferred embodiment, users are created and user rights given by the directory service, the licenses that include user tickets should be installed into the directory server. The directory service is the only service that requests user tickets and it is responsible for maintaining consistency of the used ticket counts. It periodically counts all users in the directory and their user rights and sets the number of tickets in use as appropriate.

Some kinds of tickets are "consumable" e.g. for sending mail messages, and these will not be returned unless, for example, the message is cancelled.

Clearly if an originally purchased licence becomes inadequate, due for example to an increased number of users, then supplemental licences can be purchased which when installed will increase the number of available tickets for a service. Additional functionality can of course also be purchased subsequently, in order to add new features to a system, and the appropriate software and licence installed in an appropriate server.

It is considered that with the above description of the licence management system and method proposed by the

invention, a software developer will have difficulty producing the corresponding code for licence management for a particular software product written in a particular language, and hence no further description is considered necessary in this respect.

CLAIMS

1. A software licence management method for use with a computer system including at least one server, the method being such that before a service can offer functionality to a user, the said service shall verify that the user has a licence for said service, and wherein the computer system further includes a licensing subsystem with which are associated service tickets corresponding to existing licences, the method including the steps of the said service requesting a respective service ticket from the licensing subsystem prior to offering functionality to the user, and the licensing subsystem issuing a said service ticket to the said service, if one is available, thereby verifying the licence exists and allowing the said service to offer functionality.

2. A method as claimed in Claim 1, including the step of installing licence information comprising a licence serial number and a licence activation key into the computer system, the activation key containing encoded details of the licensed services, and wherein the computer system calculates, from the serial number and the activation key, information including the types of service tickets associated with a particular licence, the numbers of service tickets, and the duration of the licence.

3. A method as claimed in Claim 2 wherein the licensing subsystem maintains a log of the numbers of the maximum available and issued service tickets.

4. A method as claimed in Claim 2 or Claim 3, wherein a covert code is calculated by the computer system from the serial number and activation key and wherein mapping of the covert code to service tickets is calculated by

the licencing subsystem each time it is started.

5. A method as claimed in any one of the preceding claims wherein the computer system comprises a plurality of computers connected in a network and wherein a said server comprises a directory server, providing a directory service and including a respective licencing subsystem, together with a directory database and a ticket database, wherein stored in the directory database are directory server profile details, licence details and user profile details, and wherein the ticket database includes details of service tickets available in accordance with the respective licence details and issued, and wherein adding a user to the computer system includes the steps of starting the directory server licensing subsystem, the directory server licensing subsystem fetching the directory server profile with licence details from the directory database and updating the ticket database, the requesting of a user service ticket by the directory service from the licensing subsystem, the checking of ticket availability in the ticket database by the licensing subsystem, the issuing of a ticket by the licensing subsystem to the directory service, and the adding to the directory database of the new user's profile by the directory service.

6. A method as claimed in any one of Claims 1 to 4 wherein the computer system comprises a plurality of computers connected in a network, wherein a said server comprises an application server providing an application service and including a respective licensing subsystem with a respective ticket database, and another said server comprises a directory server providing a directory service and with a respective directory database, wherein stored in the directory database are directory server profile details, application server profile details and licence details, and wherein the ticket

database includes details of service tickets available in accordance with the licence details and issued, and wherein obtaining a use ticket for the application service includes the steps of starting the application server licensing subsystem, the subsystem fetching the application server profile and licence details from the directory database and updating the ticket database accordingly, starting the application service without providing funtionality, the requesting by the application service of a user service ticket from the licensing subsystem, the checking of ticket availability in the ticket database by the licensing subsystem, and the issuing of a service ticket to the application service by the licensing subsystem, the application service then providing its functionality to a user.

7. A method as claimed in any one of Claims 1 to 4 and for independent licence sharing, wherein the computer system comprises a plurality of computers connected in a network, wherein a said server comprises an application server providing an application service and including a respective licensing subsystem with a respective ticket database, and another said server comprises a directory server providing a directory service and including a respective licensing subsystem with a respective ticket base and with a respective directory database, wherein stored in the directory database are directory server profile details, application server profile details and shareable licence details, the number of service tickets being unlimited, wherein the directory server ticket database includes details of service tickets available in accordance with the shareable licence details and issued, and wherein the application server ticket database includes details of service tickets issued, and wherein obtaining a service ticket for the application service includes the steps of the directory server licensing system fetching the server profile from the

directory database, analysing the shareable licence details and updating the corresponding ticket types and ticket limits in the directory server ticket database, the application server licensing subsystem fetching the application server and the directory server profiles and shareable licence details from the directory database and analysing them and updating the corresponding ticket types in the application server ticket database, starting the application service without providing functionality, the requesting by the application service of a service ticket from the application server licensing system, the granting of a service ticket, and the issuing of the service ticket to the application service by the application server licensing system, the application service then providing its functionality to a user.

8. A method as claimed in any one of Claims 1 to 4 and for licence sharing with site licensing, wherein the computer system comprises a plurality of computers connected in a network, wherein a said server comprises an application server providing an application service and including a respective licensing subsystem with a respective ticket database, another said server comprises a site licensing server providing a site licensing service and including a respective licensing subsystem with a respective ticket database, and a further said server comprises a directory server providing a directory service and having a directory database, wherein stored in the directory database are directory server profile details, site licensing server profile details, application server profile details and licence details, wherein the site licensing subsystem ticket database includes details of service tickets available in accordance with the licence details and issued, and wherein obtaining a service ticket for the application service when the application server has no

respective licence includes the steps of the licensing subsystems fetching their corresponding server profiles from the directory database, analysing the installed licence details and the site licensing server updating the respective ticket database, starting the application service without providing functionality, the requesting by the application service of a service ticket from the site licensing service, the requesting of a service ticket and a ticket-forwarding ticket by the site licensing service from its licensing subsystem, the checking of ticket availability and the issuing of the service and ticket-forwarding tickets to the site licensing service, the forwarding of the service ticket to the application server licensing subsystem, and the issuing of the service ticket to the application service, the application service then providing its functionality to a user.

9. A computer system including at least one server and a software licence management system, the management system being such that before a service can offer functionality to a user, the service shall verify that the user has a licence for said service, the management system including a licencing subsystem with which are associated service tickets corresponding to existing licences, and the management system being such that a said service ticket is issued to a service, if one is available, upon request by the service, thereby verifying existence of a licence and allowing the said service to offer functionality.

10. A computer system as claimed in Claim 9, wherein the management system includes means for calculating from an input licence serial number and input licence activation key, information including the types of service tickets associated with a particular licence, the numbers of service tickets and the duration of the licence, said

information being encoded in the activation key.

11. A computer system as claimed in Claim 10, and including a log in which are stored the numbers of the maximum available and issued service tickets.

12. A computer system as claimed in Claim 9 or Claim 10, and wherein the calculating means include means for calculating a covert code from the serial number and activation key, and the licensing subsystem including means for mapping the covert code into service tickets each time the licensing subsystem is started.

13. A computer system as claimed in any one of Claims 9 to 12 and comprising a plurality of computers connected in a network, wherein a said server comprises a directory server, providing a directory service and including a respective licensing subsystem, together with a directory database and a ticket database, wherein stored in the directory database are directory server profile details, licence details and user profile details, and wherein the ticket database includes details of service tickets available in accordance with respective licence details and issued.

14. A computer system as claimed in any one of Claims 9 to 12 and comprising a plurality of computers connected in a network, wherein a said server comprises an application server providing an application service and including a respective licensing subsystem with a respective ticket database, and another server comprises a directory server providing a directory service with a respective directory database, wherein stored in the directory database are directory server profile details, application server profile details and licence details, and wherein the ticket database includes details of service tickets available in accordance with the licence

details and issued.

15. A computer system as claimed in Claim 14 and for independent licence sharing, wherein the directory server includes a respective directory licensing subsystem and a respective ticket database, shareable licence details, for which the number of service tickets available is unlimited, being stored in the directory database, the directory ticket database including details of service tickets available in accordance with the shareable licence details and issued, and the application server ticket database including details of service tickets issued.

16. A computer system as claimed in Claim 14 and for licence sharing with site licensing, and including another said server comprising a site licensing server providing a site licensing service and including a respective licensing subsystem with a respective ticket database, the directory database also including site licensing server profile details, and wherein the site licensing ticket database includes details of service tickets available in accordance with the licence detailed and issued.

17. A software licence management method substantially as herein described with reference to an as illustrated in Figure 1, Figure 2, Figure 3, or Figure 4, of the accompanying drawings.

18. A computer system including at least one server and a software licence management system substantially as herein described with reference to and as illustrated in Figure 1, or Figure 2, or Figure 3, or Figure 4 of the accompanying drawings.

The Patent Office

29

Application No:	GB 9617596.3	**Examiner:**	Mike Davis
Claims searched:	1-18	**Date of search:**	26 September 1996

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.O): G4A (AAP)

Int Cl (Ed.6): G06F

Other:

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X	GB 2236604 A	(SUN MICROSYSTEMS)	1,9 at least
X	EP 0332304 A2	(DIGITAL EQUIPMENT)	"
X	WO 93/11480 A1	(INTERGRAPH)	"
X	US 5375206	(HUNTER ET AL)	"
X	US 4924378	(HERSHEY ET AL)	"

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

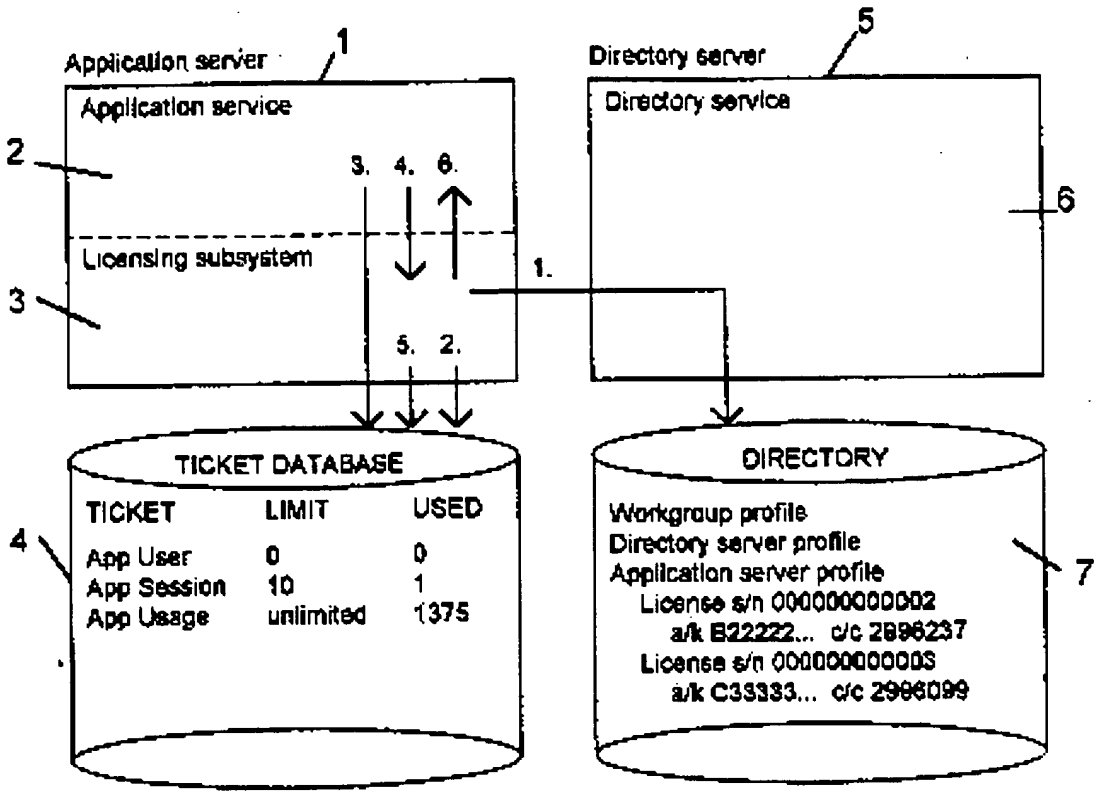
Application server
1
Application service
2
3.
4.
6.
Licensing subsystem
3
1.
5.
2.
Directory server
5
Directory service
6
TICKET DATABASE
4
TICKET
LIMIT
USED
App User
0
0
App Session
10
1
App Usage
unlimited
1375
DIRECTORY
Workgroup profile
Directory server profile
Application server profile
License s/n 000000000002
a/k B22222... c/c 2896237
License s/n 000000000003
a/k C33333... c/c 2996099
7

FIG 1

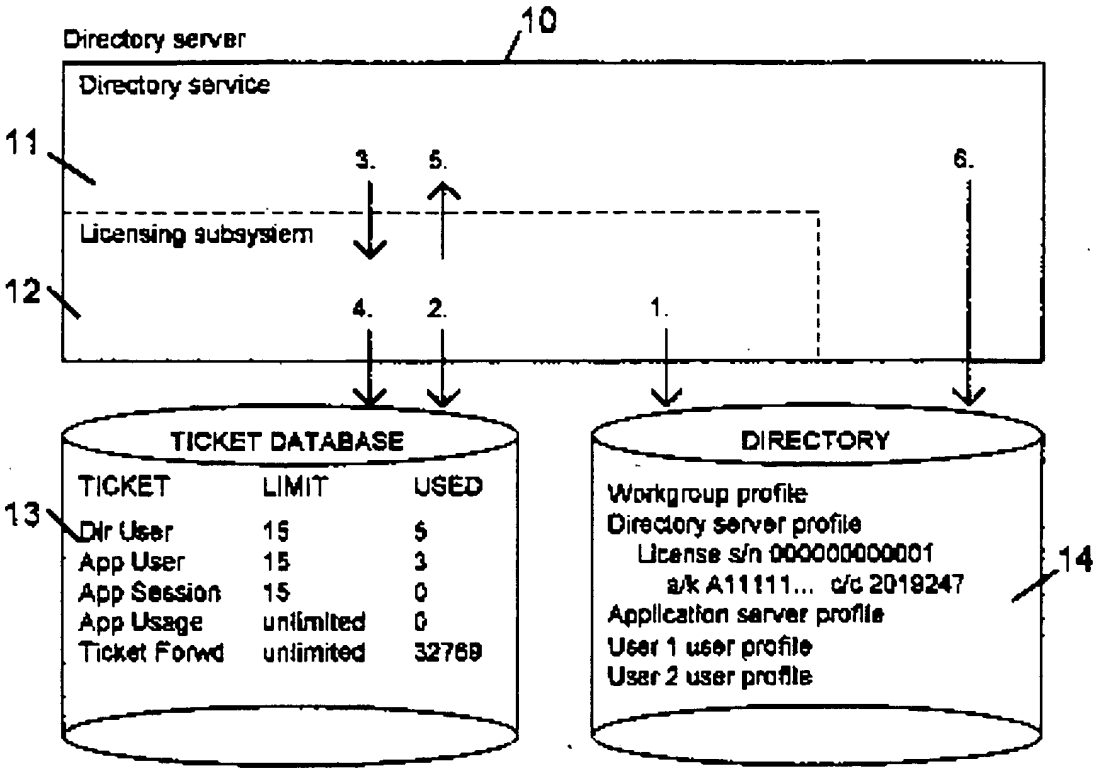
Directory server
10
Directory service
11
3.
5.
6.
Licensing subsystem
12
4.
2.
1.
TICKET DATABASE
TICKET
LIMIT
USED
13
Dir User
15
5
App User
15
3
App Session
15
0
App Usage
unlimited
0
Ticket Forwd
unlimited
32769
DIRECTORY
Workgroup profile
Directory server profile
License s/n 000000000001
a/k A11111... c/c 2019247
14
Application server profile
User 1 user profile
User 2 user profile

FIG 2

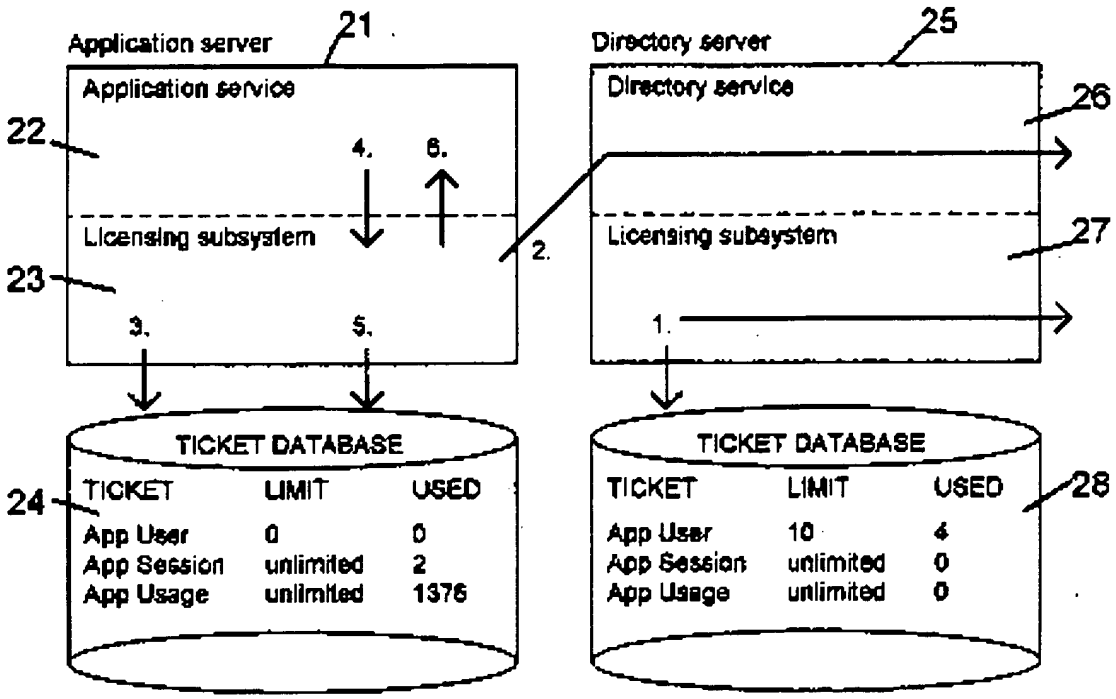
Application server
21
Application service
22
4.
6.
Licensing subsystem
23
2.
3.
5.
TICKET DATABASE
24
TICKET LIMIT USED
App User 0 0
App Session unlimited 2
App Usage unlimited 1376
Directory server
25
Directory service
26
Licensing subsystem
27
1.
TICKET DATABASE
28
TICKET LIMIT USED
App User 10 4
App Session unlimited 0
App Usage unlimited 0

FIG 3

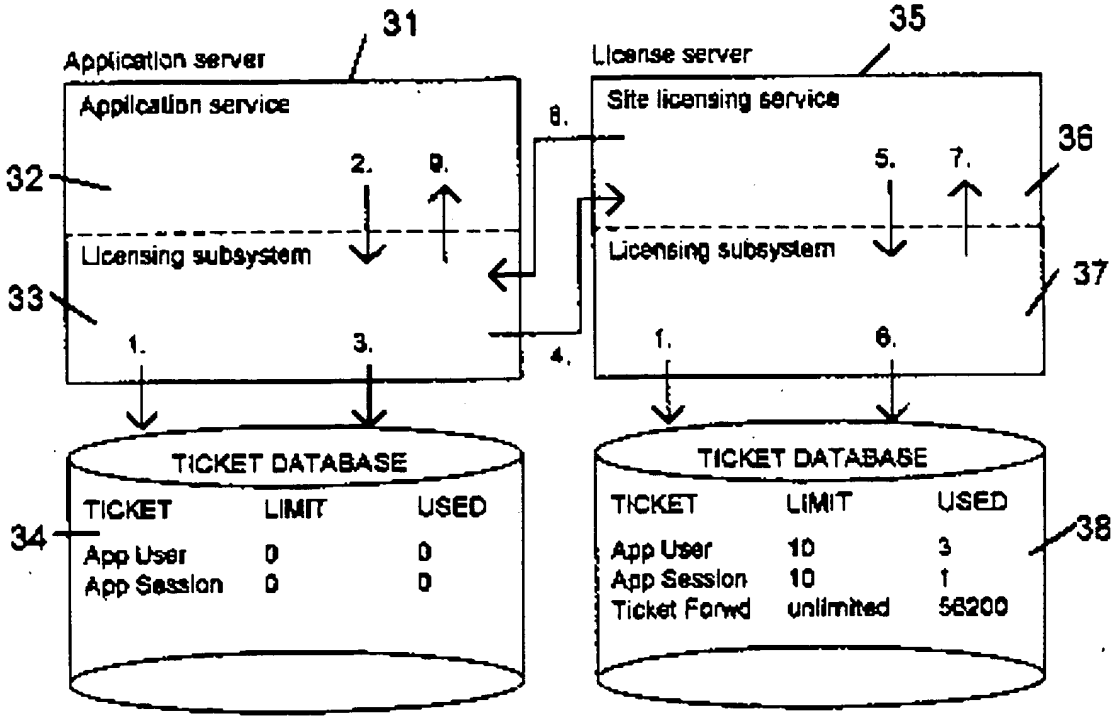

FIG 4